
Rational Access Information Plan

Send comments to: Jennifer Fell (JAF)

Comments must be returned by: May 8, 1992

Tentative start date: May 11, 1992

Approvals

All: Your signature indicates that you have read this plan and that you approve proceeding as outlined herein. Your signature also indicates that you have made this plan available to individuals in your group named in this plan, as appropriate. Any changes to this plan will require written approval of all persons

Marketing:

(Dave Bernstein, Rational Access Product Manager)

Date

Technical:

(Mike Druke, Systems and Environment Manager)

Date

Customer Support:

(Bruce Clarke, Manager of Customer Support)

Date

Production:

(Kristine Bullock, Manager of Publications Production)

Date

Documentation:

(Ellen Finch, Manager of Ada Documentation)

Date

Purpose and Scope of this Plan

This Information Plan is intended to describe everything we (the Ada Documentation Group) know about conveying information about Rational Access to its customers. This plan includes decisions, suggestions, issues, and questions about:

- The information we have about the product and which portions of this information we want to communicate to the customer.
- The information we have about the customer.
- How we want to convey this information to these customers.
- The information we have about constraints, such as schedules and resources.

Note: Because the transfer of information takes many paths (from online help to installation procedures to training) and because these paths must be complementary, this Plan attempts to describe each path, whether or not a particular path will be the responsibility of the Ada Documentation group.

Product Description

Rational Access is a Motif-style interface to the Delta version of the Rational Environment. Access significantly changes the way users perform standard operations. Specifically, standard operations are now accessible through menus and a mouse. Access includes most standard Motif conventions: mouse, pull-down menus, dialog boxes, scroll bars (when possible), buttons, and so forth. In compliance with Motif, the number of function-key bindings is significantly reduced.

Goals

Rational Access should be:

- Easy to use
- Easy to learn
- Easier to install than RXI
- Easier to sell than Rational terminals
- Inexpensive to support
- “Full service” (operations combine several low-level Environment commands)
- On the path to Rational II. Access can help users along this path by:
 - Helping them become comfortable using the mouse and menus
 - Slowly changing the way they think about and operate in the Environment.
 - Keeping them happy so they’ll want to buy Rational II.

Platforms

Rational Access is being designed to run with Sun Sparc/Sun OS and IBM RS6000/AIX as clients and any X-compatible (MIT X11R4 or greater) platforms as servers.

No other platforms are being considered at this time, although the Development group is experimenting with Xceed and PCs. The client-server setup potentially allows customers to use Access from almost any terminal or terminal emulator. Thus, it could potentially replace both RXI and RWI. For more information, see the *Rational Access Product Specification*.

The R1000 portion of Access are designed to run on Environment release D_12_6_5 (Delta 3.0) or greater, although it will run on D_12_5_0 (Delta 2.2).

Features

Because several features of Rational Access are essential to the documentation effort and fundamental to the design of the product, they are included here. This is not a complete list of features (that's what the documentation will be for).

Menu Operations and Dialog Boxes

- A set of pull-down menus is provided. The menus will be connected to a combination of existing Environment commands and newly created commands.
- Many menu operations result in the creation of dialog boxes into which parameters are entered.
 - All menu commands that require any parameters will bring up a dialog box.
 - Dialog boxes will be filled in with (editable) actual defaults for names and options. (Not strings like <SELECTION>.)
- Menu operations will be “all-purpose”; they will do everything needed to perform the operation requested.
- Operations that are unreversible will prompt for confirmation.

Buttons and Button Panel

Access will support a button panel between the pulldown menus and the Environment screen area in which users can create buttons that correspond to menu items.

- Buttons on the button panel are user-selectable.
- Buttons can be created from menu items.

(Actually, buttons can be created from anything bound to a key sequence beginning with the [Meta] key, but we probably won't document this capability for customers.)

Function Keys and Keymap

- Due to Motif requirements, the number of function keys will be radically reduced.
 - Only 12 functions keys are used.
 - If a keyboard has additional function keys, they can be defined by the user, but are not defined by Access.
 - Function keys will be reserved for operations that are performed often.
 - If a keyboard lacks certain function keys, those functions are simply not available to the user on that keyboard.

- Essential functions (not available through the mouse or menus) may be bound only to the first 9 function keys to accommodate all keyboard types.
- Function keys will be used only unmodified, [Shift]ed, and [Control]ed.
- [Control] F1 through F6 are reserved for Item keys.
- Motif reserves F1 for the Help menu and F10 for activating the menu bar.
- A single keymap and terminal type (Rational_Access) will be created for all instantiations of Access.
- Operations that appear on menus will be available through [Meta] key sequences.
- Keys will *not* be used for functions that are inconsistent with their embossed label.

Command Windows

- All existing Environment commands will still be available. Commands not found on the menus or bound to keys will be available from command windows.

Other New Features

- A new window will be provided called the Image Palette, which provides an updatable image directory. Features include:
 - The ability to list all the current images
 - The ability to cause these images to appear in Environment windows
 - The necessity to explicitly refresh the Palette
- The cursor will change to an hourglass when “Running” is displayed in the banner of the Environment message window.

Motif and Environment Overlap

Both Motif and the Environment provide windows. A single *Motif window* will contain a complete Environment screen, which may include one or more *Environment windows* (sometimes called *Environment window frames*.) The Motif window will be controlled by Motif. Environment windows will be controllable through icons and command windows.

Purpose of the Documentation

The documentation for Rational Access will probably be the only learning resource that addresses (or even makes reference to) the Access interface to the Rational Environment. Existing training and documentation are based on specific key names, logical key names, and Ada command names, almost all of which will be replaced by Access menu items. Thus,

The documentation must bridge the gap between Access and the existing documentation for the Rational Environment and the layered products. This is especially important for new users who will need a springboard into the existing User's Guide, Rational Environment Reference Manual, and online help. Without such assistance, new

users will always remain beginners. They will not be able to obtain vital information about more complex tasks.

While not the user's sole resource for information about the Environment, it will be the sole resource for information about performing operations through Access's menus, mouse, and key bindings. As such, the documentation should:

- Provide complete information for customers and Rational technical representatives (tech reps) to install Access.
- Introduce users to the use of the mouse and to the general layout of the menus. There is no need to go into great detail in either of these areas; however, some introduction is customary and useful for setting up the terms the rest of the documentation will use to discuss the tools and operations.
- Emphasize the ways in which the new interface is different from (and similar to) the ASCII "command window" interface, the Rational X-Interface (RXI), and the Rational Windows Interface (RWI). In particular, the documentation should:
 - Encourage users to explore the new menus and the mouse.
 - Position the key bindings as an alternative means of performing menu and mouse operations.
 - Not encourage the use of command windows, except for prototyping code segments, executing user-defined commands, and (in rare cases) executing Environment commands not available through the menus.
- Provide several mappings between related tasks and operations:
 - Between user tasks and all the menu, mouse, and keyboard operations for performing those tasks.
 - Between existing Environment commands and the new menu items.
 - Between Access menu items and the corresponding mouse and keyboard accelerators.

Most of these mappings will need to be multidirectional. Different users and different problems will require multiple ways of accessing the same information.

- Provide complete reference information about features specific to the Access interface, such as the behavior of each menu item and dialog box field and the ability to create buttons from menu items.
- Minimize the need for support and tech rep time. Tech reps should not have to walk each customer through Access or write supplemental guides to make their customers successful.

The documentation for Access does *not* need to emphasize the ability for customers to customize key bindings and menu items. Unlike RXI and RWI, which were positioned as very open, Access is considered a more closed, prepackaged solution. Exception: Users are encouraged to customize their workspace by creating buttons from their favorite menu items.

Audience Profile

The documentation for Rational Access could potentially address four audiences:

- Current Environment users

- New Environment users
- System administration persons
- Rational technical representatives (tech reps)

The Ada Documentation group can commit to supporting the first three of these, each at different levels. Information specifically intended for tech reps (such as a list of tested configurations) should be written by more technical persons.

My assumptions about the audiences that the Ada Documentation group needs to address are detailed below.

Current Environment Users

In the first six months that Rational Access is available, the largest group of users will be current Environment users.

What They Know

These users are familiar with standard Environment operations and logical key names. They are familiar with the item-operation paradigm. They are also familiar with the Environment library system, compilation system, and CMVC facilities. They know what command they would have executed or key they would have pressed in the past.

What They Will Want to Know

Based on conversations and late-night usability testing with Access users in the home office, current users' first questions will be, "Where is [Create Command]?" and "Where is [Promote]?" Before exploring the menus, they may spend several days continuing to use command windows. At first all they will want is something that tells them where these essential keys are.

When current users begin exploring Access, they'll need, in roughly this order:

- A *brief* introduction to the way Access takes advantage of mouse and menu capabilities, including the basics of scroll bars and dialog boxes.
- Three kinds of mappings:
 - Between tasks (explained in English) and menu items or key bindings—for when they can't remember the existing Environment command (something similar to the current *Basic Operations* book).
 - Between existing Environment commands and new menu items and key bindings—for when they know exactly what command they would have used before.
 - Between new menu items or key bindings and Environment commands—for filling in dialog boxes with options, using naming conventions like wild-cards, and for troubleshooting when the menu item doesn't do what they expected.
- Information about customizing. (Note: The ability to customize the button panel is the only customization we plan to document.)

Current Environment users might be familiar with any of several user interfaces:

- Probably the largest group is familiar with RXI.

- Depending on Access's direction, they may be familiar with RWI.
- Most are probably at least familiar with other Motif-like interfaces.
- Some users still might be upgrading from "dumb" terminals.

The RCI group has discovered that many existing Rational customers are still new to UNIX and workstations. We should be sensitive to this without redocumenting UNIX and X.

New Environment Users

Within the first year that Rational Access is generally available, we can assume that the number of new Environment users will increase rapidly, until the number of Access users is divided almost evenly between existing Environment users and new users.

What They Know

Based on the experiences of the RCI group, new users will probably more familiar with UNIX and X than our existing customers. They probably will be familiar with Motif-style user interfaces.

New users will not know anything about the Rational Environment. They probably will not be familiar with standard Environment operations or logical keys. They will not have knowledge about the Environment's formal Item-Operation paradigm or the general functionality of Common commands. They also won't be familiar with basic Environment models such as the complex library system, unit states, or subsystems.

What They Need to Know

New users will probably be more successful than existing Environment users in the first ten minutes, because they won't be approaching Access with any baggage. However, they still have the most to learn:

- What new users need foremost is Environment training. Documentation that maps user tasks onto specific operations may not have meaning for them, since many tasks and models are unique to the Environment (unit states, various types of libraries, and so on). Such documentation would become useful over time.

We have a great deal already invested in Environment documentation for new users. With the Item-Operation paradigm left intact in Access, much of this is still usable (although not ideal).

- To learn about using Access, new users will need three of the things existing Environment users will need:
 - A brief introduction to the way Access takes advantage of menu and mouse capabilities
 - A mapping between tasks (some will be familiar) and menu and mouse operations
 - Information about customizing the button panel
- So that they can use our existing Environment and layered products documents to become expert Rational users, new users will need an introduction to

the existing Environment documentation and a guide to the more useful (and less command-dependent) parts. They will definitely need some explanation of the three different levels through which they can talk to the R1000—menus, key bindings, and command windows—and how these interact.

System Administrators

System administrators may be existing Environment users or new users. They may or may not actually use Rational Access, but they will be responsible for tasks such as:

- Installing Access (R1000 and UNIX portions)
- Customizing key bindings for a site, if necessary (Note: Although possible, this is not recommended. We do not plan to document it.)
- General support for users who encounter problems

This information should be easily accessible for system administrators and should be placed where it won't confuse "mere" users. Installation may best be delivered as a separate guide that can be tossed once the product is installed (the usual way).

Approximate Breakdown of the Audiences

We can assume that in the first six months that Rational Access is available, most of its users will be existing Environment users. At one year, we can assume that the number of existing Environment users and new Environment users will begin to balance out, becoming approximately 50/50. (Based on the number of current Environment sessions being 2000-3000 and the field's commitment to sell 1000 new seats in 1992. Assuming the number of users with Suns and RS6000s will be slightly greater among the new seats.)

Issue: Waiting for information about how many existing customers will be able to use Access. Will is obsolete RXI and/or RWI. DHB is gathering this information.

Task Description

Existing Environment Tasks

The documentation for Rational Access also should enable both new and existing Environment users to perform the tasks introduced in the Fundamentals training, *Basic Operations*, and the *User's Guide*. Ideally, it should also help existing Environment users find the means through which to perform the tasks described in *Advanced Topics* training and basic CMVC operations. In general, it will provide them with instructions for performing micro-level tasks.

The documentation should enable users to determine which other documents they need and to understand the descriptions in those documents (most of which are at the "command window" level).

New Tasks

After reading the documentation for Access, users should be able to manipulate GUI- and Access-specific features. More specifically, they should be able to:

- Use the mouse to traverse the Environment library structure, manipulate Environment and Motif selections, and choose items from menus and dialog boxes.
- Use the keyboard to access menu items.
- Be able to “reason about” the menu structure and find what they need in no more than two tries.
- Create buttons from menu items.

Tasks not Documented

Because of resource and time limitations, the documentation will not be able to:

- Provide users with conceptual information about Environment models for doing things.
- Teach users about macro-level tasks, such as building an Ada program in the Rational Environment.
- Create buttons from commands not on menus (even though it is possible).

Design Implications/Documentation Deliverables

The proposed documentation design and deliverables are driven by several factors:

- Documentation resources are *extremely* limited. The Documentation group’s first two priorities for the next 6 months are documentation for Rational II (alpha in September) and RCI.
- Our primary goal is to provide documentation that is compatible with the Rational Access interface. The desire to improve Rational’s existing Environment and layered product documentation must be considered secondary.
- Because resources are limited, we need to build off of existing documentation as much as possible. Every page of new material may take as much as twice as long for us to produce. (Production time would be the same.)
- For the same reason, a single document must serve both new and existing Environment users.
- No one wants a document like the Rational X Interface manual. Everyone would prefer something that is more closely related to making users successful at using the Environment through the new interface. (Unfortunately, an RXI-like manual simply introducing the Access interface would be the easiest to produce.)
- A tutorial would be extremely writer-intensive and would require software support. Additionally, tutorials can only introduce a few simple tasks; our user’s needs will extend beyond the tutorial very quickly.

So, a proposal:

- Online documentation:
 - Reference-style help for each menu and dialog box
 - Assorted online specifications
- Printed documentation:
 - Rational Access Guide
 - Installation Procedure
 - Fundamentals Training
 - Release Information
 - Possible technical notes about tested configurations

Online

Direct Menu and Dialog Box Help

The Access menu system makes online information much more accessible and visible. These features make the online information the first priority for the Documentation group. Online help will provide the detailed reference-style information from which a user could learn to use Access if there were no other sources. Specifically, Access online help will consist of:

- A Help menu, which contains items for reference-style help on specific commands, the key bindings, mouse bindings, function keys, underlined items (explain), getting started with Access, and the other menus.
- A Help option for every dialog box.
- A special pop-up window which contains the function-key bindings. (There may be plans to allow users to click on items in this window as a means of executing the function key indicated.)
- Two Help keys:
 - Motif requires that the F1 key bring up the Help menu.
 - Environment users will want their standard Help key, which performs a What.Does operation on the object designated by the cursor. (I recommend S-F1.)

Any newly-written Help text (such as that for menus and dialog boxes) will be displayed in a pop-up Help window. The actual display within the window will be in ASCII format. Help text is intended to be brief, yet include all the information required to use the menu items or dialog box (general purpose info, brief explanation of Rational-specific concepts/usage, and parameter-level information). Each Help entry should end with a list of the related Environment commands on which the dialog box or menu item is based and for which "What.Does" help exists.

A large portion of the Access help already exists. Most of the existing help contains the appropriate level of detail and completeness. Most, however, are in need of editing for consistency with the terms used in the *User's Guide* and the *ERM*. Some dialog boxes may also need to be renamed. (2 weeks; should be someone *very* familiar with *ERM* terms and word usage.)

Online Specifications

In the usual Rational style, the following specifications will be available online:

- All key binding files and procedures (!Machine.Editor_Data.Rational_Access@)
- Package !Commands.Menu_Operations.

Existing customers, system managers, and Rational tech reps are all aware that Rational makes this kind of information visible. According to SJL's customer survey a majority of our customers actually look at the online specifications before looking for printed documentation or online help. We can assume they will continue to do the same.

Because these specifications will be visible, they should be commented appropriately. In this case of package Menu_Operations, appropriate comments should include a complete description of what each subprogram does and which already-visible (and documented) Environment subprograms it uses. Menu_Operations placement in !Commands makes such commenting even more necessary. (D3.0 Beta customers have already tripped over the package, without being told about it.)

Issue: Currently, the specification for package Menu_Operations contains subprograms that are not implemented (and may never be). Can these be removed before general availability?

Printed

Rational Access Guide

I recommend a single document, based on the existing *Basic Operations* book. A rough outline is included below, with new material in **bold**:

- Overview: An introduction to Access itself, the basic features of the interface ("Of Mice and Menus"), conventions used in the document to describe the interface elements, and an overview of how various means of using the Environment relate to each other (command windows, key bindings, menu items). Also a pointer to which of these means is documented in which other books (*ERM*, *User's Guide*, layered product guides, this guide). (20pp. 2-3 weeks)
- Basic Operations: (70pp. 6-8 weeks)
 - Logging in and Out (existing material plus **connecting to an R1000**)
 - Getting Help (existing material plus **getting help on menus and dialog boxes**)
 - Managing Windows
 - Traversing the Environment
 - Using General Editing Operations
 - Writing Text Files
 - Writing Ada Programs
 - Browsing Ada Programs
 - Debugging

- Managing Libraries
- Managing Links
- Managing Session Switches
- Managing Searchlists
- Managing Jobs
- Customizing your Workspace (existing plus **making buttons from menu items**)
- Using CMVC
- Networking

Note: Changing the outline for *Basic Operations* would be very time- and resource-intensive. We may discover obvious changes or rename certain sections, but we do *not* plan to rethink the outline.

■ **Appendixes** (1 week)

- Mapping Environment commands to menu, mouse, and keyboard operations
- Mapping menu items to Environment commands
- Keyboard overlays for function keys

(There are differing opinions on this. My feeling is that the function keys are vital enough to existing Environment users that they would appreciate an overlay. Already several people have created and printed overlays. If we don't provide them, they may become an additional tech rep task. Providing keyboard overlays on heavyweight paper makes them cheap and easy.

The outline above is one that can be used by both new and existing Environment users because it is based on the one thing they have in common: they know what (in English) they want to do.

Installation Procedure

The usual sort of Installation Procedure will be needed.

Fundamentals Training

Because Access renders the existing Fundamentals Training useless, the training will need to be rewritten. As much as possible, the scope should be restricted to changing the examples and operations. A decision will need to be made about dealing with the Item-Object paradigm. I presume a few sections will need major rewriting to remove the Item-Operation paradigm, since this is no longer the preferred means of doing things.

Release Information

The usual sort of Release Information will be needed.

Technical Notes

Semi-official but not necessarily customer-consumable notes about the client/server configurations the Development group and SMSE believe will work and/or

have tested with Access would probably be useful for tech reps. These would need to be written by one of these groups.

Media Selection

Rational Access Guide: Some people complain that the packaging of current *Basic Operations* is less than ideal for this type of document. The binder is large, difficult to use, and creates the illusion of complexity. It might be possible to present the Rational Access Guide in a slimmer, spiral-bound format. Using a heavy-weight paper for the final few pages would allow us to print multiple keyboard overlays, which users could cut out. We should explore the possibilities for doing something accessible while staying within the Delta-compatible FrameMaker template (8 1/2 x 11).

Installation Procedure: The usual 8 1/2 x 11, stapled format.

Fundamentals Training: The usual 8 1/2 x 11 format on paper and transparencies.

Release Information: The usual 8 1/2 x 11, stapled format.

Online help: Delivered and displayed in ASCII format.

Issue: We need to decide on a standard size (or sizes) for the Help windows. Users should not have to scroll right and left to read Help. (Yes, I realize users can change the size of the window. But we control the default.) I've mentioned this to CMG already.

Production Issues

Format and Tools

Most of the printed documentation will be done in FrameMaker using the Delta-compatible Ada Documentation FrameMaker template. This template enforces an 8 1/2 x 11 format similar to that of this Information Plan. FrameMaker documents definitely include:

- Rational Access Guide
- Installation Procedure (may use a similar, but not identical template?)

Some printed documentation may be done with other tools, depending on delivery requirements for these documents and whether or not they need to include screen captures. The format of these documents depend on the tools used. These documents include:

- Fundamentals Training (Powerpoint?)
- Release Note

Question: Are we delivering the Release Note online in _lpt and _ps formats? If so, we need to use Compose and cannot include screen shots (which it probably won't need anyway).

Online documentation will be done in an ASCII format (as already established and begun by the Development group) using the editor of the writer's choice.

Document Location

Printed documentation, including this *Information Plan*, will be located in /doc/env/access on the sun fileserver enterprise.

The location of online documentation to be decided when the Documentation group takes ownership of the help text from the Development group. A possible location is /doc/env/access/help on enterprise. Completed help will be delivered to customers in usr/lib/X11/app-defaults/Rational_Access/Help_Data.

Issue: We need a way to manage (CMVC-like) the Help and Rational Access Guide source. This is a general problem with everything being done in FrameMaker.

Production Concerns

We need to decide how to deliver the Rational Access Guide. It's quite long (80+ pages), which works most easily with the usual grey binders. It should be easily accessible, however, so we may want to consider something else. Spiral binding? (We need to keep is 8 1/2 x 11 because that's the size of the Delta-compatible Ada Documentation FrameMaker template.)

Documentation Concerns

Unresolved issues and project risks include:

- Documentation coordination

Recently there has been a proliferation of internal documents describing how to use Access. As soon as the Documentation group begins writing (May 15), however, we will need to take ownership of the complete documentation effort. We welcome volunteers who want to write, but we would like them to work with a writer. Working with a writer ensures that all the pieces of the documentation fit together neatly and work as a set. The success of any individual document depends on the others. The need for coordination is highlighted by the fact that many of the documents, printed and online, are being written by individuals outside the Ada Documentation group.

- Documentation project scope

Opening up the *Basic Operations* book tempts people to change the list of tasks that are included. If the project is to succeed, it *must* maintain the original list of tasks. Once the list of tasks is opened, the project will *at least* double in size.

- Interface changes

The interface must be frozen on schedule. After writing begins, any changes to the interface cause rewriting on the part of documentation. There is no time in the schedule for rewriting.

- Product goals and measurements:

The goal for the product has been described as "ease of use." A product specification exists, but is mostly a description of what the product already is, not what criteria it should meet. The teams needs to come up with something more specific—something by which we can grade the success of the product. For exam-

ple, “users should be able to tab between all fields in a dialog box.” I think everyone agrees on the rules in principle; writing them down will be a very useful exercise.

Usability testing *must* be done. Such testing will provide vital information about what problems users really have (versus the ones they remember having) and what we need to document. It also may reveal amazing things about the interface itself. The Documentation group should be involved in usability testing. (As an experiment, I sat a very experienced Rational and Motif user in front of Access and just watched him begin using it. It took him several minutes to realize he had to scroll to see help for F9-F12; he had just assumed those keys weren't bound)

- Layered products

Impact on layered product documentation has yet to be determined.

- Impact on other documentation efforts

I was originally scheduled to work on the Rational II documentation this summer. My time on Access is being taken from Rational II. There will be some fallout.

- Production Staffing

The end of the summer will be especially busy for the Production group—Rational Access, Delta 3.1, RCI, and alpha for Rational II all hit within a three-month period, each having documentation and datasheets. Access is just one of several projects added to their responsibility list that were not identified at the beginning of 1992. There needs to be a plan to allow them to be successful on each of these projects. This is a larger issue than just the Access team can resolve.

- Future

All other Delta documents reflect the command- or key-driven Environment interface. It seems likely that, since the Delta product and Access interface will be used and supported for several years, there will be pressure from customers and the field to update and reissue other Delta documents. There is no plan to do this.

Resources, Roles, and Responsibilities

Product Team

Currently, the Rational Access product team includes the following individuals. Each person's area of expertise and/or responsibility to the Documentation group is noted.

- David Bernstein, Product Manager (DHB): Providing information about the audience(s) for the documentation; ensuring that documentation pricing and distribution issues are resolved.
- Mike Druke, Development Manager (MBD): Ensuring that writers have access to the product; helping provide writers with all the technical information they need.
- Phil Levy, Product Architect (PHIL): Senior technical resource, as available.

- Developers:
 - Christina Gigante (CMG): Currently working on the R1000-driven parts of Access. These include package !Commands.Menu_Operations.
 - Brian Badenoch (BWB): Currently working on installation and the port to the RS6000.
 - Victorio Fung (VIC): Currently working on the UNIX and X.desktop parts of Access.
- Bruce Clarke, Manager of Customer Support (BCLARKE)
- Mark Dutra, SMSE (MED)
- Bob Geiger, Technical Consultant (RJG)
- Jennifer Fell, Technical Writer (JAF)

Review Team

Same as the product team. The review team also should include tech reps and other writers. So far, persons outside the product team who have committed to review the online help and Rational Access Guide include:

- Philippe Kruchten, Technical Consultant based in Santa Clara (PBK)
- Victor Ohnjec, Technical Representative based in Houston (VO)

Writing Team

The writing team, in its largest sense, includes:

- Jennifer Fell (JAF): Lead, polishing online help, writing new material for the Rational Access Guide, Release Information
- Steve Calde (SCALDE): Writing the bulk of the Rational Access guide (this is the intern; starts ramp-up May 11)
- Phil Levy (PHIL): First draft of online help
- Mark Dutra (MED): Installation Procedure
- Bob Geiger (RJG): Fundamentals Training
- ??: Online specifications, including package Menu_Operations

Production Resources

Production resources will be needed at the end of May to ensure that the Rational Access interface meets certain production standards. Specifically, the Production group has expressed an interest in:

- Editing the menu items for naming, consistency, capitalization, spelling, etc.
- Evaluating the layout of the menu items and dialog boxes; making recommendations to the Development group on related design issues.

Production resources will be needed from July through August to prepare documentation for the Beta release and again in early October for the final release:

- Rational Access Guide: editing, tweaking screen captures (there shouldn't be many) (120 hours total)
- Online help: editing (40 hours?)
- Release note: editing (30 hours)
- Installation Procedure: editing (24 hours)
- Technical notes on configurations: editing (8 hours)
- Fundamentals training: editing, tweaking many screen captures (80+ hours)

Technical Resources

See the Product Team, listed above. Because this is a new product, there are no "expert users."

Written Resources

The bulk of the outline for the Rational Access Guide will be taken directly from *Rational Environment Basic Operations*, Rational terminal edition.

Parts of the following manuals also may be used:

- The first draft of the Rational II and Sierra "corporate glossary," which will include terms for describing user interface elements and names for standard menu items. (Available no later than May 31.)
- *Rational Environment User's Guide*, Rational terminal edition
- Rational Windows Interface User's Manual
- Rational X-Interface User's Manual

Unresolved Resource Issues

Production Staffing: The end of the summer will be especially busy for the Production group—Rational Access, Delta 3.1, RCI, and alpha for Rational II all hit within a three-month period, each having documentation and datasheets. Access is just one of several projects added to their responsibility list that were not identified at the beginning of 1992. There needs to be a plan to allow them to be successful on each of these projects. This is a larger issue than just the Access team can resolve.

Schedule

As always, the documentation schedule is dependent on the software schedule. If the software schedule slips, the documentation schedule slips accordingly. Thus, all documentation deadlines should be read as "Development milestone plus # days." We won't ask you to make up our lost time, please don't ask us to make up yours.

The schedule below includes all major software and documentation milestones. Documentation milestones include milestones for the documentation produced by the Ada Documentation group and other groups such as SMSE and Training (currently subsumed by Marketing). Tracking all documentation milestones is essential to ensuring the sanity of the Production group.

Note that this schedule does not include time for the online help to be reviewed. We plan on sending out individual sections for review as we finish them.

Milestone	Responsible	Date
Information Plan review	JAF	05/01/92
Alpha testing begins (inc. field testing) (no doc)	MBD, BCLARKE, MED	05/01/92
Documentation intern arrives	ELF	05/11/92
Online help given to documentation group	PHIL (?)	05/15/92
Production review of the interface	PRODUCTION	05/25 - 06/19/92
Out of town	SCALDE (intern)	06/06 - 06/14/92
Software functional freeze (design & interface)	MBD	06/19/92
Printed documentation out for review	JAF	07/06/92 ^a
Last day for input to printed documentation	All	07/13/92
Online documentation to production group	JAF	07/13/92 ^b
Printed documentation to production group	JAF	07/20/92
Fundamentals training to production group	RJG	07/??/92 ^b
All software (inc. online help) to SMSE	MBD, JAF	08/01/92
Installation note to production group	MED	08/07/92
Release note to production group	JAF	08/07/92
Release note to SMSE	PRODUCTION	08/12/92
Beta release	DHB	08/15/92
Next rev of printed documentation to production	JAF	09/17/92 ?
Printed documentation to manufacturing	PRODUCTION	10/08/92 ^c
General availability	All	10/15/92

a. Allowing a week for review is essential; however, it really squeezes the writing time into 6 weeks (plus one week for the intern to learn the Environment and FrameMaker and one week for the intern to be out of town). This is very tight. Product manager and lead writer may need to discuss trade-offs in what is delivered for Beta versus final (e.g., index, certain sections).

b. This milestone needs to be determined and agreed upon by the writer and the Production group. We generally allow the Production group three weeks. Considering their resource constraints, they'll need to decide how to stagger the various documents.

c. This schedule allows only one week for the ECO process. It used to take six weeks. GJR has said he's planning to fix this problem. We need to be certain.